



HAL
open science

Efficient computation of large array-based mechanical structures

Augustin Parret-Fréaud, Daniel J. Rixen

► **To cite this version:**

Augustin Parret-Fréaud, Daniel J. Rixen. Efficient computation of large array-based mechanical structures. 2nd ECCOMAS Young Investigators Conference (YIC 2013), Sep 2013, Bordeaux, France. hal-00855877

HAL Id: hal-00855877

<https://hal.science/hal-00855877>

Submitted on 30 Aug 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Efficient computation of large array-based mechanical structures

Augustin Parret-Fréaud^{a,*} and Daniel J. Rixen^b

^a ENSTA ParisTech — Unité de Mécanique
828 boulevard des Maréchaux
91762 Palaiseau Cedex, France

^b Technische Universität München — Chair of Applied Mechanics
Boltzmannstr. 15
85748 Garching, Germany

*augustin.parret-freaud@ensta-paristech.fr

Abstract. *This paper deals with the simulation of mechanical structures consisting of very large arrays of unit cells. Since this special type of problem can be seen as almost-periodic, we present a strategy to analyze it as fully periodic problem together with an auxiliary one devoted to restore the finiteness of the array through the imposition of additional traction field. This enables us to speed up the resolution of such problems by extending previous works about efficient numerical methods for cyclic problems.*

Keywords: quasi-periodic structures; domain decomposition; iterative solvers.

1 INTRODUCTION

Systems made out of a high number of elementary cells arranged in arrays are more and more often encountered in engineering problem, particularly with the development of microsystem technology. Simulation of such type of systems may lead to a huge computational burden since several hundreds or thousands of microsystems are required to produce a macro-effect. Up to now, efficient techniques exist to model cyclic or infinitely extending arrays, but one crucial point is to properly take into account the fact that the array is finite. Except some previous work using, for instance, fictitious domain approaches [2], little has been done to address those type of problems.

The method discussed in this paper is an application of the domain decomposition method known as Finite Element Tearing and Interconnecting (FETI) [3, 4], developed in the last two decades as a parallel solver for very large problems. We will show how the subsequent formalism may be used to analyze array based problems as fully periodic ones, thus enabling to extend some previous work devoted to cyclic problems, based on Fourier expansion [5] or Krylov subspace enrichment [1, 6].

In the following section, we will briefly present the dual decomposition domain strategy applied to array-based problems. Then, we will describe our strategy and show how the subsequent problem may be efficiently solved.

2 SUB-STRUCTURED FORMULATION OF ARRAY-BASED PROBLEMS

For sake of simplicity, let us consider a structure (fig. 1) composed of N identical cells of same characteristics arranged in a one dimensional array. In the following, each cell (whose data will be denoted by the superscript (s)) will be considered as a sub-domain $\Omega^{(s)}$ of the initial structure Ω with left and right interfaces (whose associated data will be denoted by subscript L and R , respectively). We assume a conforming and identical finite element discretization on each cell, and denote by K the stiffness matrix (which will obviously be identical in each cell), $\mathbf{u}^{(s)}$ the displacement field and $\mathbf{f}^{(s)}$ the generalized traction field on subdomain (s) . We also introduce the interface internal forces $\lambda_{b,L}^{(s)}$ and $\lambda_{b,R}^{(s)}$ imposed on the left and right interface of $\Omega^{(s)}$ by its neighbors, respectively, and $\mathbf{t}_L^{(s)}$ and $\mathbf{t}_R^{(s)}$ the trace operators that extract from $\Omega^{(s)}$ degrees of freedom (dof) on the corresponding boundary. Then,

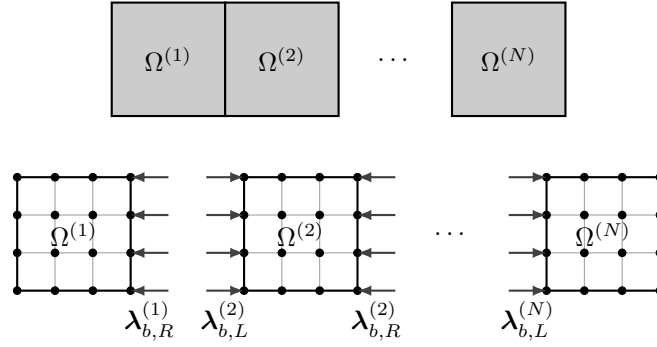


Figure 1: Array-based structure

the global mechanical problem can be expressed as a local equilibrium of each cell together with equilibrium on interfaces and compatibility condition between each cells:

$$\begin{aligned} \mathbf{K}\mathbf{u}^{(s)} &= \mathbf{f}^{(s)} + \mathbf{t}_R^{(s)} \boldsymbol{\lambda}_{b,R}^{(s)} + \mathbf{t}_L^{(s)} \boldsymbol{\lambda}_{b,L}^{(s)}, \\ \boldsymbol{\lambda}_{b,R}^{(s-1)} + \boldsymbol{\lambda}_{b,L}^{(s)} &= \mathbf{0}, \\ \mathbf{t}_R^{(s-1)} \mathbf{u}^{(s-1)} - \mathbf{t}_L^{(s)} \mathbf{u}^{(s)} &= \mathbf{0}. \end{aligned}$$

The starting point of dual domain decomposition methods (a.k.a. FETI methods [3]) is to express the reactions between subdomains in term of one unique interface traction field $\boldsymbol{\lambda}_b$ insuring the interface equilibrium. Grouping the left and right contributions of the internal forces and using classical boolean assembly operator [4], one has

$\boldsymbol{\lambda}_b^{(s)} = \begin{bmatrix} \boldsymbol{\lambda}_{b,L}^{(s)T} & \boldsymbol{\lambda}_{b,R}^{(s)T} \end{bmatrix}^T = \underline{\mathbf{A}}^{(s)} \boldsymbol{\lambda}_b$. Then, introducing $\mathbf{B}^{(s)} = \underline{\mathbf{A}}^{(s)} \mathbf{t}^{(s)}$, the previous problem may be rewritten as:

$$\sum_s \mathbf{K}\mathbf{u}^{(s)} = \mathbf{f}^{(s)} + \mathbf{B}^{(s)T} \boldsymbol{\lambda}_b, \quad \text{with} \quad \sum_s \mathbf{B}^{(s)} \mathbf{u}^{(s)} = \begin{bmatrix} \mathbf{B}_R^e & -\mathbf{B}_L^e & & \mathbf{0} \\ & \ddots & \ddots & \\ \mathbf{0} & & \mathbf{B}_R^e & -\mathbf{B}_L^e \end{bmatrix} \begin{bmatrix} \mathbf{u}^{(1)} \\ \vdots \\ \mathbf{u}^{(N)} \end{bmatrix}.$$

We now assume that all cells are non-floating, namely that there exists enough Dirichlet boundary conditions to fix the rigid body motion. In this case, the operator \mathbf{K} is non-singular so that the previous equilibrium equations can be solved and corresponding displacement substituted in the compatibility conditions. One finally obtain the well-known dual interface problem for the traction field $\boldsymbol{\lambda}_b$:

$$\mathbf{S}_d \boldsymbol{\lambda}_b = \mathbf{d}_d, \quad \text{with} \quad \mathbf{S}_d = \sum_s \mathbf{B}^{(s)} \mathbf{K} \mathbf{B}^{(s)T}.$$

In case the cells have rigid body modes, the theory can be easily extended using pseudo-inverses and projections as classical done in the FETI method.

At this stage, let's observe that the special form of operator associated to $\sum_s \mathbf{B}^{(s)} \mathbf{u}^{(s)}$ can be seen as *almost* cyclic, and so is for the interface problem since the stiffness matrix \mathbf{K} is the same for each subdomains.

In the following, we present the main points of our strategy to analyse this problem as a fully periodic one.

3 PERIODIC APPROACH

3.1 Formulation

We start by expressing the previous dual problem in a Lagrangian formalism through the following saddle point problem, whose stationary condition on $\mathbf{u}^{(s)}$ and $\boldsymbol{\lambda}_b$ enables to recover back local equilibrium and compatibility conditions:

$$\inf_{\mathbf{u}^{(s)}} \sup_{\boldsymbol{\lambda}_b} \mathcal{L}(\mathbf{u}^{(s)}, \boldsymbol{\lambda}_b), \quad \text{with} \quad \mathcal{L}(\mathbf{u}^{(s)}, \boldsymbol{\lambda}_b) = \sum_s \left(\frac{1}{2} \mathbf{u}^{(s)T} \mathbf{K} \mathbf{u}^{(s)} - \mathbf{f}^{(s)T} \mathbf{u}^{(s)} \right) - \boldsymbol{\lambda}_b^T \sum_s \mathbf{B}^{(s)} \mathbf{u}^{(s)}.$$

Next, in order to create an *artificial* cyclicity, we introduce an additional Lagrange multiplier λ_b^e between the right boundary of $\Omega^{(N)}$ and the left one of $\Omega^{(1)}$, homogeneous to a traction field, such as to link the corresponding displacements. In order to recover the original problem, this new traction field λ_b^e is enforced to vanish through the introduction of an additional Lagrange multiplier, denoted by w_{b_c} and homogeneous to a displacement field. The new (although equivalent) saddle point problem then reads:

$$\inf_{\mathbf{u}^{(s)}, \mathbf{w}_{b_c}} \sup_{\lambda_b, \lambda_b^e} \mathcal{L}_c \left(\mathbf{u}^{(s)}, \mathbf{w}_{b_c}, \lambda_b, \lambda_b^e \right);$$

with:

$$\begin{aligned} \mathcal{L}_c \left(\mathbf{u}^{(s)}, \mathbf{w}_{b_c}, \lambda_b, \lambda_b^e \right) &= \sum_s \left(\frac{1}{2} \mathbf{u}^{(s)T} \mathbf{K} \mathbf{u}^{(s)} - \mathbf{f}^{(s)} \mathbf{u}^{(s)} \right) - \lambda_b^T \sum_s \mathbf{B}^{(s)} \mathbf{u}^{(s)} \\ &\quad - \lambda_b^{eT} \left(\mathbf{t}_R^{(N)} \mathbf{u}^{(N)} - \mathbf{t}_L^{(1)} \mathbf{u}^{(1)} \right) - \mathbf{w}_{b_c}^T \lambda_b^e. \end{aligned}$$

Let's note that \mathcal{L}_c may also be expressed as:

$$\mathcal{L}_c \left(\mathbf{u}^{(s)}, \mathbf{w}_{b_c}, \lambda_b, \lambda_b^e \right) = \mathcal{L} \left(\mathbf{u}^{(s)}, \lambda_b \right) - \lambda_b^{eT} \left(\mathbf{t}_R^{(N)} \mathbf{u}^{(N)} - \mathbf{t}_L^{(1)} \mathbf{u}^{(1)} + \mathbf{w}_{b_c} \right),$$

such that this new saddle point problem is equivalent to the original one and just differs from it by the fact that the displacement gap $\mathbf{w}_{b_c} = \mathbf{t}_L^{(1)} \mathbf{u}^{(1)} - \mathbf{t}_R^{(N)} \mathbf{u}^{(N)}$ is appearing explicitly through the imposition of λ_b^e .

Now, introducing $\lambda_{b_c} = \begin{bmatrix} \lambda_b^T & \lambda_b^{eT} \end{bmatrix}^T$ defined on the extended global interface, and corresponding operator \mathbf{B}_c , the new stationary conditions on $\mathbf{u}^{(s)}$, λ_{b_c} and \mathbf{w}_{b_c} yield:

$$\begin{aligned} \mathbf{K} \mathbf{u}^{(s)} &= \mathbf{f}^{(s)} + \mathbf{B}_c^{(s)T} \lambda_{b_c}, \\ \sum_s \mathbf{B}_c^{(s)} \mathbf{u}^{(s)} + \mathbf{C}_\lambda \mathbf{w}_{b_c} &= \mathbf{0}, \\ \mathbf{C}_\lambda \lambda_{b_c} &= \mathbf{0}, \end{aligned} \quad \text{with} \quad \sum_s \mathbf{B}_c^{(s)} \mathbf{u}^{(s)} = \begin{bmatrix} \mathbf{B}_R^e & -\mathbf{B}_L^e & \mathbf{0} \\ \mathbf{0} & \ddots & \ddots \\ \mathbf{0} & \ddots & \ddots & -\mathbf{B}_L^e \\ -\mathbf{B}_L^e & \mathbf{0} & \mathbf{B}_R^e \end{bmatrix} \begin{bmatrix} \mathbf{u}^{(1)} \\ \vdots \\ \vdots \\ \mathbf{u}^{(N)} \end{bmatrix}.$$

where \mathbf{C}_λ is a restriction operator from the global interface to the external boundaries, verifying $\mathbf{C}_\lambda \lambda_{b_c} = \lambda_b^e$.

3.2 Resolution

As in section 2, equilibrium equations can be solved and corresponding displacement substituted in the compatibility conditions in such a way as to obtain new dual constrained problem on λ_{b_c} :

$$\begin{bmatrix} \mathbf{S}_{d_c}^\diamond & \mathbf{C}_\lambda^T \\ \mathbf{C}_\lambda & \mathbf{0} \end{bmatrix} \begin{pmatrix} \lambda_{b_c} \\ \mathbf{w}_{b_c} \end{pmatrix} = \begin{pmatrix} \mathbf{d}_{d_c} \\ \mathbf{0} \end{pmatrix}, \quad \text{with} \quad \mathbf{S}_{d_c}^\diamond = \sum_s \mathbf{B}_c^{(s)} \mathbf{K} \mathbf{B}_c^{(s)T}.$$

So far, let us note that the operator \mathbf{B}_c previously introduced now exhibit full cyclicity, and the same applies to $\mathbf{S}_{d_c}^\diamond$ for the same reasons as mentioned in section 2.

Previous constrained problem may now be solved in two different ways using iterative methods based on Krylov solvers [6], such as the popular conjugate gradient. First one starts by expressing \mathbf{w}_{b_c} from the previous system as:

$$\mathbf{F}_d \mathbf{w}_{b_c} = \left(\mathbf{C}_\lambda (\mathbf{S}_{d_c}^\diamond)^{-1} \mathbf{C}_\lambda^T \right) \mathbf{w}_{b_c} = \left(\mathbf{C}_\lambda (\mathbf{S}_{d_c}^\diamond)^{-1} \mathbf{d}_{d_c} \right).$$

Subsequently, a two-level algorithm may be used, in which the external loop (associated to the update of \mathbf{w}_{b_c}) will require to solve intermediate problems with $\mathbf{S}_{d_c}^\diamond$ at every iteration. A second approach consists in seeking λ_{b_c} as a solution of the following problem (where $\mathbf{P} = \mathbf{I} - \mathbf{Q} \mathbf{C}_\lambda^T (\mathbf{C}_\lambda \mathbf{Q} \mathbf{C}_\lambda^T)^{-1} \mathbf{C}_\lambda$ is a classical projector arising from the constraint $\mathbf{C}_\lambda \lambda_{b_c} = \mathbf{0}$), \mathbf{Q} being a well-chosen definite matrix:

$$(\mathbf{P}^T \mathbf{S}_{d_c}^\diamond \mathbf{P}) \lambda_{b_c}^* = \mathbf{P}^T \mathbf{d}_{d_c},$$

solved by a projected conjugate gradient algorithm preconditioned by $(\mathbf{S}_{d_c}^\diamond)^{-1}$.

Both approaches requires to inverse $\mathbf{S}_{d_c}^\diamond$ at each iteration of the external loop, allowing to use methods initially developed in the context of cyclic problems to do such inversion. The strategy used earlier for similar problems consisted in using the cyclicity of $\mathbf{S}_{d_c}^\diamond$ in order to enrich the search space at each iteration by circular permutation of search directions [1].

4 CONCLUSION

In this paper, we have presented a new strategy to efficiently solve structures constituted of elementary cells arranged in an array. Based on a domain decomposition point of view, the method allows to exhibit fully periodic problem constrained such that the original one is recovered. This allows extending some tools initially developed for cyclic problem throughout a two loops iterative algorithm based on Krylov methods, where interface traction field or displacement gap between the extremities of the structure may be equally sought. Some numerical experiments conducted on several academic problems illustrate the numerical performances of the method and will be shown at the conference.

ACKNOWLEDGEMENT

This work has been initiated during a stay of the first author at TU Delft, in the department of Precision and Microsystems Engineering, section of Engineering Dynamic then headed by the second author. The authors acknowledge the financial support for this project from the hosting department.

REFERENCES

- [1] P. Gosselet, D. J. Rixen, and C. Rey. “A domain decomposition strategy to efficiently solve structures containing repeated patterns”. In: *International Journal for Numerical Methods in Engineering* 78.7 (2009), p. 828. DOI: 10.1002/nme.2517.
- [2] A. Kaveh and H. Fazli. “Approximate eigensolution of locally modified regular structures using a substructuring technique”. In: *Computers and Structures* 89.5-6 (2011), p. 529. DOI: 10.1016/j.compstruc.2010.12.013.
- [3] C. Farhat and F. X. Roux. “Implicit parallel processing in structural mechanics”. In: *Computational Mechanics Advances* 2.1 (1994). North-Holland, pp. 1–124.
- [4] P. Gosselet and C. Rey. “Non-overlapping domain decomposition methods in structural mechanics”. In: *Archives of Computational Methods in Engineering* 13.4 (2006), p. 515. DOI: 10.1007/BF02905857.
- [5] D. J. Rixen and R. F. Lohman. “Efficient computation of eigenmodes of quasi-cyclic structures”. In: *IMAC-XXIII Conference and Exposition on Structural Dynamics*. Orlando, FL, 2003, pp. 1–13.
- [6] Y. Saad. *Iterative methods for sparse linear systems*. 3rd edition. PWS Publishing Company, 2000. ISBN: 9780898715347.